The GP debugger

B. Allombert

IMB
CNRS/Université Bordeaux 1

17/01/2013

What is it ?

```
? 1/0
  ***   at top-level: 1/0
  ***                   ^--
  *** _/_: impossible inverse in gdiv: 0.
  ***   Break loop: type 'break' to go back to GP p
break>
```

Welcome to the GP debugger !

How can I leave ?

By entering `break` or `Control-D`.

```
? 1/0
  ***   at top-level: 1/0
  ***                   ^--
  *** _/_: impossible inverse in gdiv: 0.
  ***   Break loop: type 'break' to go back to GP p
break> break

?
```

At last ! We escaped !

How can I enter ?

Three ways :

- By running a program that trigger an error
- By entering Control-C
- By using the breakpoint function

```
? mean(V)=sum(i=1,#V,V[i])/#V;
? variance(V)=mean(sqr(V))-sqr(mean(V));
? stdev(V)=sqrt(variance(V));
? stdev([1,2,3])
  ***   at top-level: stdev([1,2,3])
  ***                 ^--------------
  ***   in function stdev: sqrt(variance(V))
  ***                           ^------------
  ***   in function variance: mean(sqr(V))-sqr(mean(V))
  ***                                 ^--------------------
  *** sqr: forbidden multiplication t_VEC (3 elts) * t_V
  ***   Break loop: type 'break' to go back to GP prompt
```

This displays a call trace. Exercise : fix the program.

What can I do ?

Examine the value of variables :

```
? for(i=1,10^9,sin(i));
^C  ***   at top-level: for(i=1,10^9,sin(i))
  ***                                 ^-------
  *** sin: user interrupt after 797 ms.

  ***   Break loop: <Return> to continue; 'break' to go
break> i
162636
break>
```

So you interrupted the loop after 162636 iterations. You can continue the computation by entering Return.

```
? intnum(x=0,[0],exp(-x^2))
  ***   at top-level: intnum(x=0,[0],exp(-x^2))
  ***                                 ^----------
  *** exp: the PARI stack overflows !
  current stack size: 8000000 (7.629 Mbytes)
  [hint] you can increase GP stack with allocatemem()

  ***   Break loop: type 'break' to go back to GP prompt
break> x
-233745626852515.450341946951562160881O3
```

Something is wrong, we did not specify the end-point correctly.

Special commands

- `dbg_err` : returns the current error.
- `dbg_x` : like `\x`.
- `dbg_up, dbg_down` : go up/down one scope.

To disable the debugger : Set `breakloop=0` in your gprc.

Up and down

```
? invmod(x,p)=lift(1/Mod(x,p));
? suminv(p)=sum(i=1,p-1,invmod(i^2+1,p));
? suminv(2^32+1)
  ***   at top-level: suminv(2^32+1)
  ***                 ^--------------
  ***   in function suminv: sum(i=1,p-1,invmod(i^2+1,p))
  ***                                   ^----------------
  ***   in function invmod: lift(1/Mod(x,p))
  ***                                 ^----------
  *** _/_: impossible inverse in Fp_inv: Mod(641,4294967
  ***   Break loop: type 'break' to go back to GP prompt
break> dbg_err()
error("impossible inverse in Fp_inv: Mod(641,4294967297)
break> Vec(dbg_err())
["e_INV","Fp_inv",Mod(641,4294967297)]
```

```
break> x
23717
break> i
i
break> dbg_up()
  ***   at top-level: suminv(2^32+1)
  ***                 ^---------------
  ***   in function suminv: sum(i=1,p-1,invmod(i^2+
  ***                                     ^----------
break> i
154
```

breakpoint

breakpoint simulates pressing control-C at some point.

```
? invmod(x,p)=breakpoint();lift(1/Mod(x,p));
? suminv(p)=sum(i=1,p-1,invmod(i^2+1,p));
? suminv(7)
  ***   at top-level: suminv(7)
  ***                 ^---------
  ***   in function suminv: sum(i=1,p-1,invmod(i^2+
  ***                                   ^----------
  ***   in function invmod: breakpoint();lift(1/
  ***                       ^--------------------

  ***   Break loop: <Return> to continue; 'break' t
break> x
2
```

After entering return :

```
break>
  ***   at top-level: suminv(7)
  ***                 ^---------
  ***   in function suminv: sum(i=1,p-1,invmod(i^2+1,p))
  ***                                   ^----------------
  ***   in function invmod: breakpoint();lift(1/
  ***                       ^--------------------

  ***   Break loop: <Return> to continue; 'break' to go
break> x
5
```

```
break> dbg_up
  ***   at top-level: suminv(7)
  ***                 ^---------
  ***   in function suminv: sum(i=1,p-1,invmod(i^2+1,p))
  ***                                     ^---------------
break> i
2
```